

## REMARKS

The Office Action mailed September 18, 2008 has been carefully reviewed and the foregoing amendment has been made in consequence thereof.

Claims 1-15 are now pending in this application. Claims 1-20 stand rejected. Claims 16-20 have been cancelled.

The objection to the specification is moot in view of the cancellation of Claims 16-20. Accordingly, Applicant respectfully requests that the objection to the specification be withdrawn.

The objection to Claims 10-15 is moot in view of the amendment to Claims 10-15. Applicant has amended Claims 10-15 as suggested by the Examiner. Accordingly, Applicant respectfully requests that the objection to Claims 10-15 be withdrawn.

The objection to Claims 17-20 is moot in view of the cancellation of Claims 17-20. Accordingly, Applicant respectfully requests that the objection to Claims 17-20 be withdrawn.

The rejection of Claims 16-20 under 35 U.S.C. § 112, second paragraph, as being indefinite is respectfully traversed. Claims 16-20 have been cancelled. Accordingly, Applicant respectfully requests that the Section 112, second paragraph, rejection of Claims 16-20 be withdrawn.

The rejection of Claims 1-8 and 16-20 under 35 U.S.C. § 101 as being directed to non-statutory subject matter is respectfully traversed.

Claim 1 has been amended to recite a system that includes “a graphical user interface; and a framework . . . configured to . . . output an operational state of the framework using the graphical user interface.” Applicant submits that page 6, lines 2-4, of Applicant’s originally-filed specification supports such a recitation. Accordingly, Applicant submits that Claim 1, as amended, recites statutory subject matter in accordance with Section 101.

Claims 2-8 depend from independent Claim 1. When the recitations of Claims 2-8 are considered in combination with the recitations of Claim 1, Applicant submits that dependent Claims 2-8 likewise recite statutory subject matter in accordance with Section 101.

Claims 16-20 have been cancelled.

For at least the reasons set forth above, Applicant respectfully requests that the Section 101 rejection of Claims 1-8 and 16-20 be withdrawn.

The rejection of Claims 1-20 under 35 U.S.C. § 102(e) as being anticipated by U.S. Pat. No. 6147967 to Ying et al. (hereinafter referred to as “Ying”) is respectfully traversed.

Ying describes a distributed control network (501) that includes a bus (504) to which is connected a master node (503) and a plurality of slave nodes (523, 533) in a loop configuration. The master node (503) and each slave node (523, 533) includes a transceiver (611) connected to the bus (504) through a shunt circuit (642). The shunt circuit (642) includes a switch (644) in parallel circuit configuration with an impedance element (643). Each switch (644) is operated under control of a respective node (530) and under normal conditions remains closed, thereby allowing signals to be carried freely over the bus (504). When a fault occurs, each node (530) opens its switch (644), causing the separate portions of the bus (504) to become isolated. The master node (503) then instructs each slave node (523, 533), starting with the closest slave node (523, 533), to close its switch (644). When a slave node (523, 533) fails to respond, the fault location is detected. The master node (503) repeats the process in the opposite direction of the loop. Operation then continues, with the slave node (523, 533) nearest the fault remaining isolated. If the short circuit occurs at the master node (503), one of the slave nodes (523, 533) may take over for the master node (503) after a predetermined wait period. The slave nodes (523, 533) may be assigned to hierarchical tiers such that a first-tier slave node (523c) takes over for a first-tier master node (503), and a second-tier slave node (533c) takes over for a first-tier slave node (523). Ying also describes a fail-safe relay control circuit.

Notably, Ying does not describe or suggest, for example, an application architecture. Further, Ying does not describe or suggest a framework having a logic handler, a detector, a refresher, and a quitter, wherein the framework is configured to mediate between a middle-tier of the application architecture and an application within a front-end tier. Rather, Ying describes a master node that instructs slave nodes to close a respective switch until a faulty

slave node is isolated. Accordingly, for at least this reason, Applicant respectfully requests that the Section 102 rejection of Claims 1-20 be withdrawn.

Claim 1 recites a system including a multi-tier application architecture having a middletier, said system comprising “a graphical user interface; and a framework to mediate between an application within a front-end tier and the middletier, wherein the framework is configured to: allow the middletier to execute an object fetched by the application from a cache; when the execution of the object fails, repeatedly refresh the object within a limited number of retries; when the object refresh succeeds, return the object to the cache and again allow the middletier to execute the object; when the object refresh does not succeed within the limited number of retries, quit the application in a fail-safe way; and output an operational state of the framework using the graphical user interface.”

Ying does not describe or suggest a system including a multi-tier application architecture having a middletier as recited in Claim 1. More specifically, Ying does not describe or suggest a system that includes a framework to mediate between an application within a front-end tier and the middletier. Rather, in contrast to the present invention, Ying describes a master node that instructs slave nodes to close a respective switch until a faulty slave node is isolated. Additionally, as described in Ying, a node from a lower tier can take over for a faulty node in a higher tier to recover from a fault.

Further, Ying does not describe or suggest a system that includes a framework that mediates between a first-tier and a middletier that is configured to repeatedly refresh the object within a limited number of retries when the execution of an object fails. Rather, in contrast to the present invention, Ying describes a master node that instructs slave nodes to close a respective switch until a faulty slave node is isolated. Additionally, as described in Ying, a node from a lower tier can take over for a faulty node in a higher tier to recover from a fault.

Moreover, Ying does not describe or suggest a system that includes a framework that mediates between a first-tier and a middletier that is configured to return an object to a cache and again allow the middletier to execute the object when an object refresh succeeds, and, when the object refresh does not succeed within the limited number of retries, quit the application in a fail-safe way. Rather, in contrast to the present invention, Ying describes a master node that instructs slave nodes to close a respective switch until a faulty slave node is

isolated. Additionally, as described in Ying, a node from a lower tier can take over for a faulty node in a higher tier to recover from a fault.

Accordingly, for at least the reasons set forth above, Claim 1 is submitted to be patentable over Ying.

Claims 2-8 depend, directly or indirectly, from independent Claim 1. When the recitations of Claims 2-8 are considered in combination with the recitations of Claim 1, Applicant submits that dependent Claims 2-8 likewise are patentable over Ying.

Claim 9 recites a method of executing an application, said method comprising “transmitting an object used by the application within a first tier to a second tier; executing a logic program at the second tier, wherein the logic program corresponds to the transmitted object; detecting an execution status of the logic program at the first tier, said detecting comprising: detecting when the execution of the logic program fails such that the object becomes stale; repeatedly refreshing the object within a limited number of retries; if said refreshing succeeds, then returning the object to the first tier and transmitting a second object to the second tier from the first tier; and if said refreshing does not succeed within the limited number of retries, then quitting the application in a fail-safe way.”

Ying does not describe or suggest a method of executing an application as recited in Claim 9. More specifically, Ying does not describe or suggest a method that includes detecting an execution status of a logic program within a second tier at a first tier. Rather, in contrast to the present invention, Ying describes a master node that instructs slave nodes to close a respective switch until a faulty slave node is isolated. Additionally, as described in Ying, a node from a lower tier can take over for a faulty node in a higher tier to recover from a fault.

Further, Ying does not describe or suggest a method that includes detecting an execution status of a logic program within a second tier at the first tier, wherein the detecting includes repeatedly refreshing an object within a limited number of retries. Rather, in contrast to the present invention, Ying describes a master node that instructs slave nodes to close a respective switch until a faulty slave node is isolated. Additionally, as described in Ying, a node from a lower tier can take over for a faulty node in a higher tier to recover from a fault.

Moreover, Ying does not describe or suggest a method that includes detecting an execution status of a logic program within a second tier at the first tier, wherein the detecting includes returning an object to the first tier and transmitting a second object to the second tier from the first tier if a refreshing succeeds, and if the refreshing does not succeed within a limited number of retries, quitting the application in a fail-safe way. Rather, in contrast to the present invention, Ying describes a master node that instructs slave nodes to close a respective switch until a faulty slave node is isolated. Additionally, as described in Ying, a node from a lower tier can take over for a faulty node in a higher tier to recover from a fault.

Accordingly, for at least the reasons set forth above, Claim 9 is submitted to be patentable over Ying.

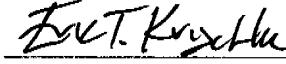
Claims 10-15 depend, directly or indirectly, from independent Claim 9. When the recitations of Claims 10-15 are considered in combination with the recitations of Claim 9, Applicant submits that dependent Claims 10-15 likewise are patentable over Ying.

Claims 16-20 have been cancelled.

For at least the reasons set forth above, Applicant respectfully requests that the Section 102 rejection of Claims 1-20 be withdrawn.

In view of the foregoing amendment and remarks, all the claims now active in this application are believed to be in condition for allowance. Reconsideration and favorable action are respectfully solicited.

Respectfully submitted,

  
Eric T. Krischke  
Reg. No. 42,769  
ARMSTRONG TEASDALE LLP  
One Metropolitan Square, Suite 2600  
St. Louis, Missouri 63102-2740  
(314) 621-5070